This listing of claims will replace all prior versions, and listings, of claims in the application.

Listing of Claims:

1 -5. (Canceled)

6. (Currently amended) The method of claim 31 further comprising:
providing a cursor associated with the ~~on any type of~~ query ~~executed~~.

7. (Canceled)

8. (Currently amended) The method of claim 31 further comprising:
[[the]] marshaling [[of]] data between the runtime environment and the database management system ~~an unmanaged layer and a managed layer~~.

9. (Currently amended) The method of claim 31, further comprising:
running ~~wherein~~ an application operation selected from a group of operations comprising functions, procedures, and triggers is ~~executed~~ directly in the database management system [[DBMS]].

10.-15. (Canceled)

16. (Currently amended) The system of claim 37, wherein the memory further comprises instructions that upon execution cause the computer system to:
provide ~~further comprising a subsystem for providing~~ a cursor for the ~~on any type of~~ query ~~executed~~.

17. (Canceled)

18. (Currently amended) The system of claim 37, wherein the memory further comprises instructions that upon execution cause the computer system to:

marshal ~~further comprising a subsystem for the marshaling of~~ data between the runtime environment and the database management system ~~an unmanaged layer and a managed layer~~.

19. (Currently amended) The system of claim 37, wherein the memory further comprises instructions that upon execution cause the computer system to:

run ~~further comprising a subsystem for~~ an application operation selected from a group of operations comprising functions, procedures, and triggers encoded in the native query language ~~to be executed~~ directly in the [[DBMS]] database management system.

20.-25. (Canceled)

26. (Currently amended) The computer-readable storage medium ~~instructions~~ of claim 43 further comprising instructions that upon execution cause the computer system to:

~~for providing~~ provide a cursor on any type of query executed.

27. (Canceled)

28. (Currently amended) The computer-readable storage medium ~~instructions~~ of claim 43 further comprising instructions that upon execution cause the computer system to:

~~for the marshaling of~~ marshal data between the runtime environment and the database management system ~~an unmanaged layer and a managed layer~~.

29. (Currently amended) The computer-readable storage medium ~~instructions~~ of claim 43 further comprising instructions that upon execution cause the computer system to:

run [[for]] an application operation selected from a group of operations comprising functions, procedures, and triggers encoded in the native query language ~~to be executed~~ directly in the database management system [[DBMS]].

30. (Canceled)

31. (Currently amended) A computer-implemented method, ~~for executing .NET managed code in a database management system (DBMS) having a database server, the method~~ comprising:

running, on a computer system, a database management system, the database management system configured to process queries generated in a native query language for the database management system;

running, on the computer system, a runtime environment configured to mange execution of intermediate language code;

establishing a connection to a client, the client associated with a set of database access privileges;

receiving, from a client, a query that invokes intermediate language code;

~~executing instructions from a memory in the database server invoking .NET managed code;~~

compiling by the runtime environment during the runtime of the database management system, the intermediate language code into an expression encoded in the native query language;

generating a context object including ~~invoking an invocation context in the database server, wherein the invocation context is based on at least a context class, wherein the context class includes~~ information comprising the set of database access privileges, a connection context of [[a]] the client, a command context of the client, a transaction context of the client, a pipe context of the client, and a trigger context of the client;

~~separating the .NET managed code into an immutable part and a mutable part;~~

exposing the context [[class]] object to the database management system ~~server~~ through the utilization of an in-process provider, wherein the in-process provider keeps track of ~~unmanaged~~ data obtained from the database management system that is referenced from the runtime environment ~~a managed space~~ and prevents access of the database management data ~~unmanaged data~~ outside a runtime environment ~~managed~~ execution frame;

executing the expression encoded in the native query language based on the context object ~~the .NET managed code in the database server based on the invocation context and the separation into immutable and mutable parts~~; and

storing information indicative of a result of the query ~~for the context class~~ in ~~said~~ memory.

32. – 35. (Canceled)

36. (Currently amended) The method of claim 31, wherein the in-process provider supports more than one pending executing command for [[a]] the connection [[of]] to the client.

37. (Currently amended) A system for executing application code in a database management system (DBMS) ~~comprising a processor and a memory~~, the system comprising:

a processor; and

a memory coupled to the processor, the memory including instructions stored therein that upon execution cause the processor to:

~~at least one processor comprising:~~

run a database management system, the database management system configured to process queries generated in a native query language;

run a runtime environment configured to mange execution of intermediate language code;

establishing a connection to a client, the client associated with a set of database access privileges;

receive a query that invokes intermediate language code from the client;

compile by the runtime environment, the intermediate language code into an expression encoded in the native query language;

generate a context object including ~~a subsystem for invoking .NET managed code and an invocation context in the database server, wherein the invocation context is based on at least a context class, wherein the context class contains~~ information comprising the

set of database access privileges, a connection context of [[a]] the client, a command context of the client, a transaction context of the client, a pipe context of the client, and a trigger context of the client;

~~a subsystem for separating the .NET managed code into an immutable part and a mutable part;~~

expose ~~a subsystem for exposing~~ the context [[class]] object to the database management system [[server]] through the utilization of an in-process provider, wherein the in-process provider keeps track of ~~unmanaged~~ data obtained from the database management system that is referenced from the runtime environment ~~a managed space~~ and prevents access of the database management data ~~unmanaged data~~ outside a ~~managed~~ runtime environment execution frame;

~~a subsystem for executing the .NET managed code in the database server~~ execute the expression encoded in the native query language based on the ~~invocation~~ context object ~~and the separation into immutable and mutable parts, wherein the code is executed under the client's connection context; and~~

~~a computing memory communicatively coupled to the processor, the computing memory operable to store information for the client's connection context~~.

38. (Currently amended) The system of claim 37, wherein the context object ~~invocation context~~ further comprises:

a command with a state execution context;

the transaction context of [[a]] the client associated with a command;

a path through which requests and results may be sent or received between the client and the database management system ~~server~~; and

a forward-only cursor on top of statement execution results.

39. (Canceled)

40. (Canceled)

41. (Canceled)

42. (Previously presented) The system of claim 37, wherein the in-process provider supports for more than one pending executing command for a connection of the client.

43. (Currently amended) A computer-readable storage medium comprising computer-readable instructions stored thereon that upon execution by a processor of a computer system cause the computer system to ~~for executing application code in a database management system (DBMS), the computer-readable instructions comprising instructions for~~:

run a database management system, the database management system configured to process queries generated in a native query language;

run a runtime environment configured to mange execution of intermediate language code;

establishing a connection to a client, the client associated with a set of database access privileges;

receive a query that invokes intermediate language code from the client;

compile by the runtime environment, the intermediate language code into an expression encoded in the native query language;

~~receiving application code, rewritten as .NET managed code, from an application;~~

generate a context object including ~~invoking .NET managed code and an invocation context in the database server, wherein the invocation context is based on at least a context class, wherein the context class contains information comprising~~ the set of database access privileges, a connection context of a client, a command context of the client, a transaction context of the client, a pipe context of the client, and a trigger context of the client;

~~separating the .NET managed code into an immutable part and a mutable;~~

~~exposing~~ expose the context [[class]] object to the database management system [[server]] through the utilization of an in-process provider, wherein the in-process provider keeps track of ~~unmanaged~~ data obtained from the database management system that is referenced from

the runtime environment ~~a managed space~~ and prevents access of the database management data ~~unmanaged data~~ outside a runtime environment ~~managed~~ execution frame; and

execute the expression encoded in the native query language ~~executing the .NET managed code in the database server~~ based on the ~~invocation~~ context object ~~and the separation into immutable and mutable parts~~.

44. (Currently amended) The computer-readable instructions of claim 43, wherein context object ~~exposing the invocation context~~ further comprises ~~exposing at least one of~~:

a command with a state execution context;

a transaction context associated with a command;

a path through which requests and results may be sent or received between the client and the database management system ~~server~~;

a trigger context, wherein the trigger results from an operation of the client; [[or]] and

a forward-only cursor on top of statement execution results.

45. (Canceled)

46. (Canceled)

47. (Canceled)

48. (Currently amended) The computer-readable storage medium ~~instructions~~ of claim 43, wherein the in-process provider supports more than one pending executing command for a connection of the client.